Amendments to the Drawings

Block 810 of **Fig. 8** is amended to correct a typographical error (changing "deparment's" to "department's"). No new matter is introduced with this amendment.

Serial No. 09/827,290   -8-   Docket RSW920000173US1

REMARKS

Claims 1, 5 - 7, 10 - 11, and 14 have been amended. Claims 3 - 4, 9, and 13 have been cancelled from the application without prejudice. Claims 15 - 18 have been added. No new matter is added with these amendments, which are supported in the specification as originally filed. Claims 1, 5 - 7, 10 - 11, and 14 - 18 are now in the application.

I.   Proposed Correction to the Drawings

A proposed replacement drawing is submitted herewith for Fig. 8. The correction made in this replacement drawing is discussed above in "Amendments to the Drawings". No new matter is introduced with the replacement drawing.

II.   Rejection Under 35 U.S.C. §102(b)

Paragraph 9 of the Office Action dated December 22, 2004 (hereinafter, "the Office Action") states that Claims 1, 3, 4, 6, 7, 9, 11, and 13 are rejected under 35 U.S.C. §102(b) as being anticipated by Rubin (U. S. Patent 5,412,797). Claims 3 - 4, 9, and 13 have been cancelled from the application without prejudice. This rejection is respectfully traversed with regard to remaining Claims 1, 6 - 7, and 11.

Applicants' independent Claims 1, 7, and 11 have been amended herein to more clearly specify limitations pertaining to associations and the ends thereof. In particular, Claim 1 (for example) specifies that an association end "reflects an association from an instance of a first class to an instance of a second class" (line 4) and that the association has "an inverse association end

Serial No. 09/827,290   -9-   Docket RSW920000173US1

... to reflect an inverse association from the instance of the second class to the instance of the first class" (lines 10 - 11 and 19 - 20). Applicants' specification uses the term "bi-directional links" to characterize this scenario where a association has an inverse, and can be navigated in both directions (that is, _from_ the instance of the first class _to_ the instance of the second class, as described in line 4 of Claim 1 and vice versa, as described in lines 10 - 11 and 19 - 20 of Claim 1); see p. 15, lines 1 - 4.

Rubin teaches techniques whereby "relations" are maintained using _pointers_ in source instances and in sink instances. See, for example, col. 2, lines 46 - 51. In particular, a _source_ instance points to the (1) first and (2) last sink instances in a ring of sink instances (col. 2 line 48), and a _sink_ instance points to (1) the next sink instance in this ring, (2) the previous sink instance in this ring, and (3) the associated source instance (col. 2, lines 49 - 51 and lines 52 - 57).

In a scenario where there are 3 or more sink instances, Rubin's source instances _do not point to_ all of the sink instances -- rather, his source instances point only to 2 of those sink instances (that is, to the "first" sink instance and the "last" sink instance in a ring, as noted above). Furthermore, Rubin specifically notes, at col. 18, line 43 - col. 19, line 2, that his invention can be employed such that his relationships can be inserted or removed _without_ requiring access to source instances. This is because each sink instance stores _only a pointer_ to its source instance. Thus, using the procedure discussed in this cited text, a new sink instance must always be inserted such that it is _neither_ the _first sink_ nor the _last sink_. This restriction is

Serial No. 09/827,290   -10-   Docket RSW920000173US1

necessary because the source instance _does_ need to be updated if its first sink or last sink pointers are changed. The cited text therefore states that the source instance can be initialized to use "dummy" sink instances as the first and last sink instances of a ring, and thereafter all "real" sink instances are inserted in between those dummy instances (see, in particular, col. 18, lines 65 – 67). Using this approach, only the _next sink_ pointers and _previous sink_ pointers of _sink_ instances in the ring need to change: the newly-inserted sink instance merely includes a reference (i.e., pointer) to the previously-created source instance. Thus, there is no modification of the "source end" of Rubin's relationship (see col. 19, lines 1 – 2, stating that neither read nor write access to the source instance is required; see also col. 19, lines 48 – 51, stating that Rubin's relationships can "sometimes" be removed without having read or write access to the source instance -- provided, apparently, that the approach of using dummy instances has been followed).

This is in contrast to Applicants' claimed invention, whereby _each end_ of an association is changed when _either_ of the ends is to be set -- that is, the end to be set _and its inverse_ are _both_ changed. See the second and third limitations of Applicants' independent claims, where in the second limitation, "setting" steps are specified for both of the association ends and in the third limitation, an "adding" and a "setting" step are specified.

Applicants further respectfully note that the analysis presented on page 4, line 12 - page 5, line 2 of the Office Action has misinterpreted "single multiplicity", as that term is used in Applicants' invention. In particular, Rubin's "presents" relationship has a _many_ multiplicity, not a _single_ multiplicity. That is because the source of a "presents" relationship, according to

Serial No. 09/827,290   -11-   Docket RSW920000173US1

Rubin's example, may have <u>many</u> related sink instances. This is similar to Applicants' "employees" association end, where the source of the "employees" association -- namely, a particular Department instance -- may have many associations to particular Employee instances. In Rubin's example, the "isPresentedBy" relationship is the one that has the <u>single</u> multiplicity -- because, as in Applicants' example whereby a particular Employee instance has a "department" association with at most one Department instance, each of Rubin's sink instances "isPresentedBy" at most one source instance. Thus, in contrast to the statement on p. 4, lines 14 - 16 of the Office Action, Rubin's "presents" relationship has the "many" multiplicity precisely because it can be associated to, or *presents*, more than one sink instance.

In addition, the statement on p. 4, lines 9 - 10 of the Office Action misinterprets this single-multiplicity and many-multiplicity concept. As stated therein, Rubin's one-to-many relationships have both single and many multiplicity. As the "single multiplicity" and "many multiplicity" terms are used in Applicants' invention, by contrast, they consider only the <u>upper bound</u> of the multiplicity. For example, a <u>one-to-many</u> (or zero-to-many) association end has <u>only</u> the <u>many</u> multiplicity. Its inverse, if and only if it is a <u>one-to-one</u> association end, has the <u>single</u> multiplicity. This is illustrated in Applicants' Fig. 2, where association end 210, for the "department" association end, is denoted as "1..1" -- that is, a one-to-one association end -- whereas association end 220, for the "employees" association end, is denoted as "0..*" -- that is, a zero-to-many association end. Page 4, lines 3 - 4 and lines 7 - 8 of Applicants' specification refer to these "1..1" and "0..*" notations as indicating the multiplicity of the association ends; examples provided in Applicants' specification then use the terms "single" and "many"

Serial No. 09/827,290   -12-   Docket RSW920000173US1

multiplicity, respectively. (Note that each end of Applicants' associations are directional: that is, the claimed "request to set an association end" reflects an association from the instance of the first class to the instance of the second class, and the claimed "inverse association end" reflects an association from the instance of the second class to the instance of the first class.)

In view of the above, Applicants respectfully submit that their independent Claims 1, 7, and 11 are patentable over Rubin. Dependent Claim 6 is therefore deemed patentable over Rubin as well. Accordingly, Applicants respectfully request that the Examiner withdraw the §102 rejection.

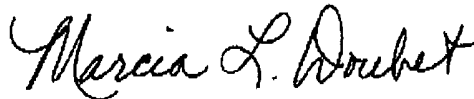III.   Rejection Under 35 U.S.C. §103(a)

Paragraph 11 of the Office Action states that Claims 5, 10, and 14 are rejected under 35 U.S.C. §103(a) as being unpatentable over Rubin in view of Johnson (a publication from javaworld.com). This rejection is respectfully traversed.

As demonstrated above, Rubin fails to render Applicants' independent Claims 1, 7, and 11 unpatentable. Rubin and Johnson therefore cannot be combined (assuming, *arguendo*, that one of skill in the art would be motivated to attempt such combination and that such combination can, in fact, be made) to render Applicants' dependent Claims 5, 10, and 14 obvious. Accordingly, Applicants respectfully request that the Examiner withdraw the §103 rejection.

Serial No. 09/827,290   -13-   Docket RSW920000173US1

IV.   Conclusion

Applicants respectfully request reconsideration of the pending rejected claims,

withdrawal of all presently outstanding rejections, and allowance of all remaining claims at an

early date.

Respectfully submitted,

Marcia L. Doubet
Attorney for Applicants
Reg. No. 40,999

Customer Number for Correspondence: 43168
Phone: 407-343-7586
Fax: 407-343-7587

Attachment: Replacement Sheet (1)

Serial No. 09/827,290   -14-   Docket RSW920000173US1